An electronic device displays one or more user interface objects on a touch-sensitive display. The device detects a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects, and determines whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. While the predefined condition is satisfied, the device displays a first menu of a plurality of menus; detects a rotation of the finger contact; and replaces display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.
Figure 1A
Portable Multifunction Device 100

210 is SIM card slot
212 is headphone jack

Figure 2
Device 300

Memory 370

Operating System 126
Communication Module 128
Contact/Motion Module 130
Graphics Module 132
Text Input Module 134

Applications

Contacts Module 137
Telephone Module 138
Video Conference Module 139
E-mail Client Module 140
Instant Messaging Module 141
Workout Support Module 142
Camera Module 143
Image Management Module 144
Video Player Module 145
Music Player Module 146
Browser Module 147
Calendar Module 148

Widget Modules

Weather Widget 149-1
Stocks Widget 149-2
Calculator Widget 149-3
Alarm Clock Widget 149-4
Dictionary Widget 149-5
User-Created Widget(s) 149-6

Widget Creator Module 150
Search Module 151
Drawing Module 152
Presentation Module 153
Word Processing Module 154
Website Creation Module 155
Disk Authoring Module 156
Spreadsheet Module 157

Device/Global Internal State 157
Portable Multifunction Device 100

- Speaker 111
- Optical Sensor 164
- Proximity Sensor 166

Current Time

- IM
- Text 141
- Calendar 148
- Photos 144
- Camera 143
- Online Video 155
- Stocks 149-2
- Map 154
- Weather 149-1
- Clock 149-4
- Calculator 149-3
- Notes 153
- Settings 412

- Phone 138
- Mail 140
- Browser 147
- iPod 152

Figure 4B
Figure 5A
600

Display one or more user interface objects on a touch-sensitive display

602

Detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects

604

Determine whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display.

606

The predefined condition includes that the finger contact is made with a finger of a non-dominant hand

608

The predefined condition includes that the finger contact is made with a thumb of a non-dominant hand

610

While the predefined condition is satisfied:

612

Display a first menu of a plurality of menus

614

Detect a rotation of the finger contact (see Figure 6B)

616

Replace display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact (see Figure 6C)

618

Detect a movement of the finger contact across the touch-sensitive display; and move the second menu in accordance with the movement of the finger contact

620

Detect a second rotation of the finger contact; and replace display of the second menu with a third menu of the plurality of menus in accordance with the second rotation of the finger contact

622

Detect activation of an operation in the second menu and, in response, perform a corresponding operation

624

Figure 6A
Detect a rotation of the finger contact

Determine a touch area that corresponds to the finger contact on the touch-sensitive display

Determine a major axis of the touch area. Detecting the rotation of the finger contact includes detecting a rotation of the major axis of the touch area.

Determining a minor axis of the touch area. Detecting the rotation of the finger contact includes detecting a rotation of the minor axis of the touch area.

Figure 6B
Replace display of the first menu with a second menu of the plurality of menus in accordance with the rotation of the finger contact.

Displaying the second menu includes displaying the second menu at a respective angle in accordance with the rotation of the finger contact.

Displaying the second menu includes displaying the second menu at a respective radial location relative to a location of the finger contact in accordance with the rotation of the finger contact.

Displaying the second menu of the plurality of menus includes displaying the second menu of the plurality of menus independent of a location of the finger contact on the touch-sensitive display.

Displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact includes displaying the second menu of the plurality of menus at a location of the finger contact.

Displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact includes displaying the second menu of the plurality of menus proximate to the location of the finger contact.

Displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact includes displaying the second menu of the plurality of menus at a central location of the touch-sensitive display.

Figure 6C
DEVICE, METHOD, AND GRAPHICAL USER INTERFACE FOR DISPLAYING MENUS

TECHNICAL FIELD

[0001] This relates generally to electronic devices with touch-sensitive surfaces, including but not limited to electronic devices with touch-sensitive surfaces that display a plurality of menus.

BACKGROUND

[0002] The use of touch-sensitive surfaces as input devices for computers and other electronic computing devices has increased significantly in recent years. Exemplary touch-sensitive surfaces include touch pads and touch screen displays. Such surfaces are widely used to manipulate user interface objects on a display.

[0003] Exemplary manipulations include changing the attributes (e.g., colors, line thickness, shading, etc.) of one or more user interface objects. Exemplary user interface objects include digital images, video, text, icons, and other graphics. A user may need to perform such manipulations on user interface objects in an operating system graphical user interface (e.g., the user interface for Mac OS X from Apple Inc. of Cupertino, Calif.), a file management program (e.g., Finder from Apple Inc. of Cupertino, Calif.), an image management application (e.g., Aperture or iPhoto from Apple Inc. of Cupertino, Calif.), a digital content (e.g., videos and music) management application (e.g., iTunes from Apple Inc. of Cupertino, Calif.), a drawing application, a presentation application (e.g., Keynote from Apple Inc. of Cupertino, Calif.), a word processing application (e.g., Pages from Apple Inc. of Cupertino, Calif.), a website creation application (e.g., iWeb from Apple Inc. of Cupertino, Calif.), a disk authoring application (e.g., iDVD from Apple Inc. of Cupertino, Calif.), or a spreadsheet application (e.g., Numbers from Apple Inc. of Cupertino, Calif.).

[0004] Graphical user interfaces often use menus to display operations available to a user, such as commands that may be used to manipulate one or more user interface objects. For example, manipulation of user interface objects often requires an activation of a tool or a formatting function (e.g., colors, line thickness, shading, etc.), which is often displayed as a menu icon in a menu.

[0005] But existing methods for displaying menus are cumbersome and inefficient. For example, existing keyboard and mouse based methods for displaying menus often use multiple steps to display various menus and sub-menus, which are tedious and create a significant cognitive burden on a user. In addition, existing menu display methods take longer than necessary, thereby wasting energy. This latter consideration is particularly important in battery-operated devices.

SUMMARY

[0006] Accordingly, there is a need for computing devices with faster, more efficient methods and interfaces for displaying menus. Such methods and interfaces may complement or replace conventional methods for displaying menus. Such methods and interfaces reduce the cognitive burden on a user and produce a more efficient human-machine interface. For battery-operated computing devices, such methods and interfaces conserve power and increase the time between battery charges.

[0007] The above deficiencies and other problems associated with user interfaces for computing devices with touch-sensitive surfaces are reduced or eliminated by the disclosed devices. In some embodiments, the device is a desktop computer. In some embodiments, the device is portable (e.g., a notebook computer, tablet computer, or handheld device). In some embodiments, the device has a touchpad. In some embodiments, the device has a touch-sensitive display (also known as a "touch screen" or "touch screen display"). In some embodiments, the device has a graphical user interface (GUI), one or more processors, memory and one or more modules, programs or sets of instructions stored in the memory for performing multiple functions. In some embodiments, the user interacts with the GUI primarily through finger contacts and gestures on the touch-sensitive surface. In some embodiments, the functions may include image editing, drawing, presenting, word processing, website creating, disk authoring, spreadsheet making, game playing, telephoning, video conferencing, e-mailing, instant messaging, workout support, digital photographing, digital videoing, web browsing, digital music playing, and/or digital video playing. Executable instructions for performing these functions may be included in a computer readable storage medium or other computer program product configured for execution by one or more processors.

[0008] In accordance with some embodiments, an electronic device includes: a touch-sensitive display, one or more processors, memory, and one or more programs. The one or more programs are stored in the memory and configured to be executed by the one or more processors. The one or more programs include instructions for: displaying one or more user interface objects on the touch-sensitive display; detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects; and determining whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. The one or more programs also include instructions for, while the predefined condition is satisfied: displaying a first menu of a plurality of menus; detecting a rotation of the finger contact; and replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

[0009] In accordance with some embodiments, a method is performed at an electronic device with a touch-sensitive display. The method includes: displaying one or more user interface objects on the touch-sensitive display; detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects; and determining whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. The method also includes, while the predefined condition is satisfied: displaying a first menu of a plurality of menus; detecting a rotation of the finger contact; and replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

[0010] In accordance with some embodiments, a computer readable storage medium has stored therein instructions which, when executed by an electronic device with a touch-sensitive display, cause the device to: display one or more user interface objects on the touch-sensitive display; detect a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects; and
determine whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. The instructions when executed also cause the device to, while the predefined condition is satisfied: display a first menu of a plurality of menus; detect a rotation of the finger contact; and replace display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

[0011] In accordance with some embodiments, a graphical user interface on an electronic device with a touch-sensitive display, a memory, and one or more processors to execute one or more programs stored in the memory includes one or more user interface objects on the touch-sensitive display. A finger contact is detected on the touch-sensitive display at a first location away from any of the displayed user interface objects. It is determined whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. While the predefined condition is satisfied: a first menu of a plurality of menus is displayed; a rotation of the finger contact is detected; and display of the first menu is replaced with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

[0012] In accordance with some embodiments, an electronic device includes: a touch-sensitive display; menus for displaying one or more user interface objects on the touch-sensitive display; means for detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects; and means for determining whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. The electronic device also includes, while the predefined condition is satisfied: means for displaying a first menu of a plurality of menus; means for detecting a rotation of the finger contact; and means for replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

[0013] In accordance with some embodiments, an information processing apparatus for use in an electronic device with a touch-sensitive display includes: means for displaying one or more user interface objects on the touch-sensitive display; means for detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects; and means for determining whether a predefined condition is satisfied. The predefined condition includes that the finger contact continues to be detected on the touch-sensitive display. The information processing apparatus also includes, while the predefined condition is satisfied: means for displaying a first menu of a plurality of menus; means for detecting a rotation of the finger contact; and means for replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

[0014] Thus, electronic devices with touch-sensitive displays are provided with faster, more efficient methods and interfaces for displaying menus, thereby increasing the effectiveness, efficiency, and user satisfaction with such devices. Such methods and interfaces may complement or replace conventional methods for displaying menus.

**BRIEF DESCRIPTION OF THE DRAWINGS**

[0015] For a better understanding of the aforementioned embodiments of the invention as well as additional embodiments thereof, reference should be made to the Description of Embodiments below, in conjunction with the following drawings in which like reference numerals refer to corresponding parts throughout the figures.

[0016] FIGS. 1A and 1B are block diagrams illustrating portable multifunction devices with touch-sensitive displays in accordance with some embodiments.

[0017] FIG. 1C is a block diagram illustrating exemplary components for event handling in accordance with some embodiments.

[0018] FIG. 2 illustrates a portable multifunction device having a touch screen in accordance with some embodiments.

[0019] FIG. 3 is a block diagram of an exemplary multifunction device with a display and a touch-sensitive surface in accordance with some embodiments.

[0020] FIGS. 4A and 4B illustrate exemplary user interfaces for a menu of applications on a portable multifunction device in accordance with some embodiments.

[0021] FIG. 4C illustrates an exemplary user interface for a multifunction device with a touch-sensitive surface that is separate from the display in accordance with some embodiments.

[0022] FIGS. 5A-5J illustrate exemplary user interfaces for displaying menus in accordance with some embodiments.

[0023] FIGS. 6A-6C are flow diagrams illustrating a method of displaying menus in accordance with some embodiments.

**DESCRIPTION OF EMBODIMENTS**

[0024] Many electronic devices have graphical user interfaces with a large number of menus. Menus show operations available to a user, such as commands that may be used to manipulate one or more user interface objects. For example, manipulation of user interface objects often requires activation of a tool or a formatting function (e.g., colors, line thickness, shading, etc.), which is often displayed as a menu item in a menu. In the embodiments described below, an improved method for displaying menus is disclosed. A finger contact is detected at a location away from any displayed user interface objects. While a predefined condition is satisfied (e.g., a thumb contact is detected), a first menu is displayed. Detecting rotation of the thumb contact replaces the first menu with a second menu. Detecting further rotation of the thumb contact replaces the second menu with a third menu, and so on. This method streamlines the menu display process by displaying a menu in response to detecting a contact that does not correspond to a particular user interface object, thereby eliminating the need for touching the location of a menu object. This method also streamlines the menu selection process by displaying a series of distinct menus in response to detecting a rotation of the contact, thereby eliminating the need to translate the contact or use multiple contacts to view multiple menus.

[0025] Below, FIGS. 1A-1C, 2, and 3 provide a description of exemplary devices. FIGS. 4A-4C and 5A-5J illustrate exemplary user interfaces for displaying menus and manipulating user interface objects. FIGS. 6A-6C are flow diagrams illustrating a method of displaying menus. The user interfaces in FIGS. 5A-5J are used to illustrate the processes in FIGS. 6A-6C.

[0026] Reference will now be made in detail to embodiments, examples of which are illustrated in the accompanying drawings. In the following detailed description, numerous specific details are set forth in order to provide a thorough
understanding of the present invention. However, it will be apparent to one of ordinary skill in the art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, circuits, and networks have not been described in detail so as not to unnecessarily obscure aspects of the embodiments.

[0027] It will also be understood that, although the terms first, second, etc. may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first contact could be termed a second contact, and, similarly, a second contact could be termed a first contact, without departing from the scope of the present invention. The first contact and the second contact are both contacts, but they are not the same contact.

[0028] The terminology used in the description of the invention herein is for the purpose of describing particular embodiments only and is not intended to be limiting of the invention. As used in the description of the invention and the appended claims, the singular forms “a”, “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term “and/or” as used herein refers to and encompasses any and all possible combinations of one or more of the associated listed items. It will be further understood that the terms “includes,” “including,” “comprises,” and/or “comprising,” when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0029] As used herein, the term “if” may be construed to mean “when” or “upon” or “in response to determining” or “in response to detecting,” depending on the context. Similarly, the phrase “if it is determined” or “if [a stated condition or event] is detected” may be construed to mean “upon determining” or “in response to determining” or “upon detecting [the stated condition or event]” or “in response to detecting [the stated condition or event],” depending on the context.

[0030] As used herein, the term “resolution” of a display refers to the number of pixels (also called “pixel counts” or “pixel resolution”) along each axis or in each dimension of the display. For example, a display may have a resolution of 320x480 pixels. Furthermore, as used herein, the term “resolution” of a multifunction device refers to the resolution of a display in the multifunction device. The term “resolution” does not imply any limitations on the size of each pixel or the spacing of pixels. For example, compared to a first display with a 1024x768-pixel resolution, a second display with a 320x480-pixel resolution has a lower resolution. However, it should be noted that the physical size of a display depends not only on the pixel resolution, but also on many other factors, including the pixel size and the spacing of pixels. Therefore, the first display may have the same, smaller, or larger physical size, compared to the second display.

[0031] As used herein, the term “video resolution” of a display refers to the density of pixels along each axis or in each dimension of the display. The video resolution is often measured in a dots-per-inch (DPI) unit, which counts the number of pixels that can be placed in a line within the span of one inch along a respective dimension of the display.

[0032] Embodiments of computing devices, user interfaces for such devices, and associated processes for using such devices are described. In some embodiments, the computing device is a portable communications device, such as a mobile telephone, that also contains other functions, such as PDA and/or music player functions. Exemplary embodiments of portable multifunction devices include, without limitation, the iPhone®, iPod Touch®, and iPad® devices from Apple Inc. of Cupertino, Calif. Other portable devices, such as laptops or tablet computers with touch-sensitive surfaces (e.g., touch screen displays and/or touch pads), may also be used. It should also be understood that, in some embodiments, the device is not a portable communications device, but is a desktop computer with a touch-sensitive surface (e.g., a touch screen display and/or a touch pad).

[0033] In the discussion that follows, a computing device that includes a display and a touch-sensitive surface is described. It should be understood, however, that the computing device may include one or more other physical user-interface devices, such as a physical keyboard, a mouse and/or a joystick.

[0034] The device supports a variety of applications, such as one or more of the following: a drawing application, a presentation application, a word processing application, a website creation application, a disk authoring application, a spreadsheet application, a gaming application, a telephone application, a video conferencing application, an e-mail application, an instant messaging application, a workout support application, a photo management application, a digital camera application, a digital video camera application, a web browsing application, a digital music player application, and/or a digital video player application.

[0035] The various applications that may be executed on the device may use at least one common physical user-interface device, such as the touch-sensitive surface. One or more functions of the touch-sensitive surface as well as corresponding information displayed on the device may be adjusted and/or varied from one application to the next and/or within a respective application. In this way, a common physical architecture (such as the touch-sensitive surface) of the device may support the variety of applications with user interfaces that are intuitive and transparent to the user.

[0036] The user interfaces may include one or more soft keyboard embodiments. The soft keyboard embodiments may include standard (QWERTY) and/or non-standard configurations of symbols on the displayed icons of the keyboard, such as those described in U.S. patent application Ser. Nos. 11/459,606, “Keyboards For Portable Electronic Devices,” filed Jul. 24, 2006, and 11/459,615, “Touch Screen Keyboards For Portable Electronic Devices,” filed Jul. 24, 2006; the contents of which are hereby incorporated by reference in their entireties. The keyboard embodiments may include a reduced number of icons (or soft keys) relative to the number of keys in existing physical keyboards, such as that for a typewriter. This may make it easier for users to select one or more icons in the keyboard, and thus, one or more corresponding symbols. The keyboard embodiments may be adaptive. For example, displayed icons may be modified in accordance with user actions, such as selecting one or more icons and/or one or more corresponding symbols. One or more applications on the device may utilize common and/or different keyboard embodiments. Thus, the keyboard embodiment used may be tailored to at least some of the applications. In some embodiments, one or more keyboard embodiments may be tailored to a respective user. For example, one or more keyboard embodiments may be tailored to a respective user
based on a word usage history (lexicography, slang, individual usage) of the respective user. Some of the keyboard embodiments may be adjusted to reduce a probability of a user error when selecting one or more icons, and thus one or more symbols, when using the soft keyboard embodiments.

[0037] Attention is now directed toward embodiments of portable devices with touch-sensitive displays. FIGS. 1A and 1B are block diagrams illustrating portable multifunction devices 100 with touch-sensitive displays 112 in accordance with some embodiments. Touch-sensitive display 112 is sometimes called a "touch screen" for convenience, and may also be known as or called a touch-sensitive display system. Device 100 may include memory 102 (which may include one or more computer readable storage mediums), memory controller 122, one or more processing units (CPU’s) 120, peripherals interface 118, RF circuitry 108, audio circuitry 110, speaker 111, microphone 113, input/output (I/O) sub-system 106, other input or control devices 116, and external port 124. Device 100 may include one or more optical sensors 164. These components may communicate over one or more communication buses or signal lines 103.

[0038] It should be appreciated that device 100 is only one example of a portable multifunction device, and that device 100 may have more or fewer components than shown, may combine two or more components, or may have a different configuration or arrangement of the components. The various components shown in FIGS. 1A and 1B may be implemented in hardware, software, or a combination of both hardware and software, including one or more signal processing and/or application specific integrated circuits.

[0039] Memory 102 may include high-speed random access memory and may also include non-volatile memory, such as one or more magnetic disk storage devices, flash memory devices, or other non-volatile solid-state memory devices. Access to memory 102 by other components of device 100, such as CPU 120 and the peripherals interface 118, may be controlled by memory controller 122.

[0040] Peripherals interface 118 can be used to couple input and output peripherals of the device to CPU 120 and memory 102. The one or more processors 120 run or execute various software programs and/or sets of instructions stored in memory 102 to perform various functions for device 100 and to process data.

[0041] In some embodiments, peripherals interface 118, CPU 120, and memory controller 122 may be implemented on a single chip, such as chip 104. In some other embodiments, they may be implemented on separate chips.

[0042] RF (radio frequency) circuitry 108 receives and sends RF signals, also called electromagnetic signals. RF circuitry 108 converts electrical signals to/from electromagnetic signals and communicates with communications networks and other communications devices via the electromagnetic signals. RF circuitry 108 may also include well-known circuitry for performing these functions, including but not limited to an antenna system, an RF transceiver, one or more amplifiers, a tuner, one or more oscillators, a digital signal processor, a CODEC chipset, a subscriber identity module (SIM) card, memory, and so forth. RF circuitry 108 may communicate with networks, such as the Internet, also referred to as the World Wide Web (WWW), an intranet and/or a wireless network, such as a cellular telephone network, a wireless local area network (LAN) and/or a metropolitan area network (MAN), and other devices by wireless communication. The wireless communication may use any of a plurality of communications standards, protocols and technologies, including but not limited to Global System for Mobile Communications (GSM), Enhanced Data GSM Environment (EDGE), high-speed downlink packet access (HS- DPA), high-speed uplink packet access (HSUPA), wideband code division multiple access (W-CDMA), code division multiple access (CDMA), time division multiple access (TDMA), Bluetooth, Wireless Fidelity (Wi-Fi) (e.g., IEEE 802.11a, IEEE 802.11b, IEEE 802.11g and/or IEEE 802.11n), voice over Internet Protocol (VoIP), Wi-MAX, a protocol for e-mail (e.g., Internet message access protocol (IMAP) and/or post office protocol (POP)), instant messaging (e.g., extensible messaging and presence protocol (XMPP), Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions (SIMPLE), Instant Messaging and Presence Service (IMPS), and/or Short Message Service (SMS), or any other suitable communication protocol, including communication protocols not yet developed as of the filing date of this document.

[0043] Audio circuitry 110, speaker 111, and microphone 113 provide an audio interface between a user and device 100. Audio circuitry 110 receives audio data from peripherals interface 118, converts the audio data to an electrical signal, and transmits the electrical signal to speaker 111. Speaker 111 converts the electrical signal to human-audible sound waves. Audio circuitry 110 also receives electrical signals converted by microphone 113 from sound waves. Audio circuitry 110 converts the electrical signal to audio data and transmits the audio data to peripherals interface 118 for processing. Audio data may be retrieved from and/or transmitted to memory 102 and/or RF circuitry 108 by peripherals interface 118. In some embodiments, audio circuitry 110 also includes a headset jack (e.g., 212, FIG. 2). The headset jack provides an interface between audio circuitry 110 and removable audio input/output peripherals, such as output-only headphones or a headset with both output (e.g., a headphone for one or both ears) and input (e.g., a microphone).

[0044] I/O subsystem 106 couples input/output peripherals on device 100, such as touch screen 112 and other input control devices 116, to peripherals interface 118. I/O subsystem 106 may include display controller 156 and one or more input controllers 160 for other input or control devices. The one or more input controllers 160 receive/send electrical signals from/to other input or control devices 116. The other input control devices 116 may include physical buttons (e.g., push buttons, rocker buttons, etc.), dials, slider switches, joy-sticks, click wheels, and so forth. In some alternate embodiments, input controller(s) 160 may be coupled to any (or none) of the following: a keyboard, infrared port, USB port, and a pointer device such as a mouse. The one or more buttons (e.g., 208, FIG. 2) may include an up/down button for volume control of speaker 111 and/or microphone 113. The one or more buttons may include a push button (e.g., 206, FIG. 2). A quick press of the push button may disengage a lock of touch screen 112 or begin a process that uses gestures on the touch screen to unlock the device, as described in U.S. patent application Ser. No. 11/322,549, "Unlocking a Device by Performing Gestures on an Unlock Image," filed Dec. 23, 2005, which is hereby incorporated by reference in its entirety. A longer press of the push button (e.g., 206) may turn power to device 100 on or off. The user may be able to customize a functionality of one or more of the buttons. Touch screen 112 is used to implement virtual or soft buttons and one or more soft keyboards.
Touch-sensitive display 112 provides an input interface and an output interface between the device and a user. Display controller 156 receives and/or sends electrical signals from/to touch screen 112. Touch screen 112 displays visual output to the user. The visual output may include graphics, text, icons, video, and any combination thereof (collectively termed "graphics"). In some embodiments, some or all of the visual output may correspond to user-interface objects.

Touch screen 112 has a touch-sensitive surface, sensor or set of sensors that accepts input from the user based on haptic and/or tactile contact. Touch screen 112 and display controller 156 (along with any associated modules and/or sets of instructions in memory 102) detect contact (and any movement or breaking of the contact) on touch screen 112 and converts the detected contact into interaction with user-interface objects (e.g., one or more soft keys, icons, web pages or images) that are displayed on touch screen 112. In an exemplary embodiment, a point of contact between touch screen 112 and the user corresponds to a finger of the user.

Touch screen 112 may use LCD (liquid crystal display) technology, LPD (light emitting polymer display) technology, or LED (light emitting diode) technology, although other display technologies may be used in other embodiments. Touch screen 112 and display controller 156 may detect contact and any movement or breaking thereof using any of a plurality of touch sensing technologies now known or later developed, including but not limited to capacitive, resistive, infrared, and surface acoustic wave technologies, as well as other proximity sensor arrays or other elements for determining one or more points of contact with touch screen 112. In an exemplary embodiment, projected mutual capacitance sensing technology is used, such as that found in the iPhone®, iPod® Touch, and iPod® from Apple Inc. of Cupertino, Calif.

A touch-sensitive display in some embodiments of touch screen 112 may be analogous to the multi-touch sensitive touchpads described in the following U.S. Pat. Nos. 6,323,846 (Westerman et al.), 6,570,557 (Westerman et al.), and/or 6,677,932 (Westerman), and/or U.S. Patent Publication 2002/0015024A1, each of which is hereby incorporated by reference in its entirety. However, touch screen 112 displays visual output from portable device 100, whereas touch sensitive touchpads do not provide visual output.


Touch screen 112 may have a video resolution in excess of 100 dpi. In some embodiments, the touch screen has a video resolution of approximately 160 dpi. The user may make contact with touch screen 112 using any suitable object or appendage, such as a stylus, a finger, and so forth. In some embodiments, the user interface is designed to work primarily with finger-based contacts and gestures, which can be less precise than stylus-based input due to the larger area of contact of a finger on the touch screen. In some embodiments, the device translates the rough finger-based input into a precise pointer/cursor position or command for performing the actions desired by the user.

In some embodiments, in addition to the touch screen, device 100 may include a touchpad (not shown) for activating or deactivating particular functions. In some embodiments, the touchpad is a touch-sensitive area of the device that, unlike the touch screen, does not display visual output. The touchpad may be a touch-sensitive surface that is separate from touch screen 112 or an extension of the touch-sensitive surface formed by the touch screen.

In some embodiments, device 100 may include a physical or virtual wheel (e.g., a click wheel) as input control device 116. A user may navigate among and interact with one or more graphical objects (e.g., icons) displayed in touch screen 112 by rotating the click wheel or by moving a point of contact with the click wheel (e.g., where the amount of movement of the point of contact is measured by its angular displacement with respect to a center point of the click wheel). The click wheel may also be used to select one or more of the displayed icons. For example, the user may press down on at least a portion of the click wheel or an associated button. User commands and navigation commands provided by the user via the click wheel may be processed by input controller 160 as well as one or more of the modules and/or sets of instructions in memory 102. For a virtual click wheel, the click wheel and click wheel controller may be part of touch screen 112 and display controller 156, respectively. For a virtual click wheel, the click wheel may be either an opaque or semitransparent object that appears and disappears on the touch screen display in response to user interaction with the device. In some embodiments, a virtual click wheel is displayed on the touch screen of a portable multifunction device and operated by user contact with the touch screen.

Device 100 also includes power system 162 for powering the various components. Power system 162 may include a power management system, one or more power sources (e.g., battery, alternating current (AC)), a recharge system, a power failure detection circuit, a power converter or inverter, a power status indicator (e.g., a light-emitting diode (LED)) and any other components associated with the generation, management and distribution of power in portable devices.

Device 100 may also include one or more optical sensors 164. FIGS. 1A and 1B show an optical sensor coupled to optical sensor controller 158 in I/O subsystem 106. Optical sensor 164 may include charge-coupled device (CCD) or complementary metal-oxide semiconductor (CMOS) phototransistors. Optical sensor 164 receives light from the environment, projected through one or more lens, and converts the light to data representing an image. In conjunction with imaging module 143 (also called a camera module), optical sensor 164 may capture still images or video. In some embodiments, an optical sensor is located on the back of device 100, opposite touch screen display 112 on the front of the device, so that
the touch screen display may be used as a viewfinder for still and/or video image acquisition. In some embodiments, an optical sensor is located on the front of the device so that the user’s image may be obtained for videoconferencing while the user views the other video conference participants on the touch screen display. In some embodiments, the position of optical sensor 164 can be changed by the user (e.g., by rotating the lens and the sensor in the device housing) so that a single optical sensor 164 may be used along with the touch screen display for both video conferencing and still and/or video image acquisition.

[0055]  Device 100 may also include one or more proximity sensors 166. FIGS. 1A and 1B show proximity sensor 166 coupled to peripherals interface 118. Alternately, proximity sensor 166 may be coupled to input controller 160 in I/O subsystem 106. Proximity sensor 166 may perform as described in U.S. patent application Ser. Nos. 11/241,839, “Proximity Detector In Handheld Device”; 11/240,788, “Proximity Detector In Handheld Device”; 11/620,702, “Using Ambient Light Sensor To Augment Proximity Sensor Output”; 11/586,862, “Automated Response To And Sensing Of User Activity In Portable Devices”; and 11/638,251, “Methods And Systems For Automatic Configuration Of Peripherals,” which are hereby incorporated by reference in their entirety. In some embodiments, the proximity sensor turns off and disables touch screen 112 when the multifunction device is placed near the user’s ear (e.g., when the user is making a phone call).

[0056]  Device 100 may also include one or more accelerometers 168. FIGS. 1A and 1B show accelerometer 168 coupled to peripherals interface 118. Alternately, accelerometer 168 may be coupled to an input controller 160 in I/O subsystem 106. Accelerometer 168 may perform as described in U.S. Patent Publication No. 20050190059, “Acceleration-based Theft Detection System for Portable Electronic Devices,” and U.S. Patent Publication No. 20060017692, “Methods And Apparatuses For Operating A Portable Device Based On An Accelerometer,” both of which are incorporated by reference herein in their entirety. In some embodiments, information is displayed on the touch screen display in a portrait view or a landscape view based on an analysis of data received from the one or more accelerometers. Device 100 optionally includes, in addition to accelerometer(s) 168, a magnetometer (not shown) and a GPS (or GLONASS or other global navigation system) receiver (not shown) for obtaining information concerning the location and orientation (e.g., portrait or landscape) of device 100.

[0057]  In some embodiments, the software components stored in memory 102 include operating system 126, communication module (or set of instructions) 128, contact/motion module (or set of instructions) 130, graphics module (or set of instructions) 132, text input module (or set of instructions) 134, Global Positioning System (GPS) module (or set of instructions) 135, and applications (or set of instructions) 136. Furthermore, in some embodiments memory 102 stores device/global internal state 157, as shown in FIGS. 1A, 1B and 3. Device/global internal state 157 includes one or more of: active application state, indicating which applications, if any, are currently active; display state, indicating what applications, views or other information occupy various regions of touch screen display 112; sensor state, including information obtained from the device’s various sensors and input control devices 116; and location information concerning the device’s location and/or attitude.

[0058]  Operating system 126 (e.g., Darwin, RTXC, LINUX, UNIX, OS X, WINDOWS, or an embedded operating system such as VxWorks) includes various software components and/or drivers for controlling and managing general system tasks (e.g., memory management, storage device control, power management, etc.) and facilitates communication between various hardware and software components.

[0059]  Communication module 128 facilitates communication with other devices over one or more external ports 124 and also includes various software components for handling data received by RF circuitry 108 and/or external port 124. External port 124 (e.g., Universal Serial Bus (USB), FIREWIRE, etc.) is adapted for coupling directly to other devices or indirectly over a network (e.g., the Internet, wireless LAN, etc.). In some embodiments, the external port is a multi-pin (e.g., 30-pin) connector that is the same as, or similar to and/or compatible with the 30-pin connector used on iPod (trademark of Apple Inc.) devices.

[0060]  Contact/motion module 130 may detect contact with touch screen 112 (in conjunction with display controller 156) and other touch sensitive devices (e.g., a touchpad or physical click wheel). Contact/motion module 130 includes various software components for performing various operations related to detection of contact, such as determining if contact has occurred (e.g., detecting a finger-down event), determining if there is movement of the contact and tracking the movement across the touch-sensitive surface (e.g., detecting one or more finger-dragging events), and determining if the contact has ceased (e.g., detecting a finger-up event or a break in contact). Contact/motion module 130 receives contact data from the touch-sensitive surface. Determining movement of the point of contact, which is represented by a series of contact data, may include determining speed (magnitude), velocity (magnitude and direction), and/or an acceleration (a change in magnitude and/or direction) of the point of contact. These operations may be applied to single contacts (e.g., one finger contacts) or to multiple simultaneous contacts (e.g., “multitouch”/“multiple finger contacts”). In some embodiments, contact/motion module 130 and display controller 156 detects contact on a touchpad. In some embodiments, contact/motion module 130 and controller 160 detects contact on a click wheel.

[0061]  Contact/motion module 130 may detect a gesture input by a user. Different gestures on the touch-sensitive surface have different contact patterns. Thus, a gesture may be detected by detecting a particular contact pattern. For example, detecting a finger tap gesture includes detecting a finger-down event followed by detecting a finger-up (lift off) event at the same position (or substantially the same position) as the finger-down event (e.g., at the position of an icon). As another example, detecting a finger swipe gesture on the touch-sensitive surface includes detecting a finger-down event followed by detecting one or more finger-dragging events, and subsequently followed by detecting a finger-up (lift off) event.

[0062]  Graphics module 132 includes various known software components for rendering and displaying graphics on touch screen 112 or other display, including components for changing the intensity of graphics that are displayed. As used herein, the term “graphics” includes any object that can be displayed to a user, including without limitation text, web pages, icons (such as user-interface objects including soft keys), digital images, videos, animations and the like.
In some embodiments, graphics module 132 stores data representing graphics to be used. Each graphic may be assigned a corresponding code. Graphics module 132 receives, from applications etc., one or more codes specifying graphics to be displayed along with, if necessary, coordinate data and other graphic property data, and then generates screen image data to output to display controller 156.

Text input module 134, which may be a component of graphics module 132, provides soft keyboards for entering text in various applications (e.g., contacts 137, e-mail 140, IM 141, browser 147, and any other application that needs text input).

GPS module 135 determines the location of the device and provides this information for use in various applications (e.g., to telephone 138 for use in location-based dialing, to camera 143 as picture/video metadata, and to applications that provide location-based services such as weather widgets, local yellow page widgets, and map/navigation widgets).

Applications 136 may include the following modules (or sets of instructions), or a subset or superset thereof:

- Contacts module 137 (sometimes called an address book or contact list);
- Telephone module 138;
- Video conferencing module 139;
- E-mail client module 140;
- Instant messaging (IM) module 141;
- Workout support module 142;
- Camera module 143 for still and/or video images;
- Image management module 144;
- Video player module 145;
- Music player module 146;
- Browser module 147;
- Calendar module 148;
- Widget modules 149, which may include one or more of: weather widget 149-1, stocks widget 149-2, calculator widget 149-3, alarm clock widget 149-4, dictionary widget 149-5, and other widgets obtained by the user, as well as user-created widgets 149-6;
- Widget creator module 150 for making user-created widgets 149-6;
- Search module 151;
- Video and music player module 152, which merges video player module 145 and music player module 146;
- Notes module 153;
- Map module 154; and/or
- Online video module 155.

Examples of other applications 136 that may be stored in memory 102 include other word processing applications, other image editing applications, drawing applications, presentation applications, JAVA-enabled applications, encryption, digital rights management, voice recognition, and voice replication.

In conjunction with touch screen 112, display controller 156, contact module 130, graphics module 132, and text input module 134, contacts module 137 may be used to manage an address book or contact list (e.g., stored in application internal state 192 of contacts module 137 in memory 102 or memory 370), including: adding name(s) to the address book; deleting name(s) from the address book; associating telephone number(s), e-mail address(es), physical address(es) or other information with a name; associating an image with a name; categorizing and sorting names; providing telephone numbers or e-mail addresses to initiate and/or facilitate communications by telephone 138, video conference 139, e-mail 140, or IM 141; and so forth.

In conjunction with RF circuitry 108, audio circuitry 110, speaker 111, microphone 113, touch screen 112, display controller 156, contact module 130, graphics module 132, and text input module 134, telephone module 138 may be used to enter a sequence of characters corresponding to a telephone number, access one or more telephone numbers in address book 137, modify a telephone number that has been entered, dial a respective telephone number, conduct a conversation and disconnect or hang up when the conversation is completed. As noted above, the wireless communication may use any of a plurality of communications standards, protocols and technologies.

In conjunction with RF circuitry 108, audio circuitry 110, speaker 111, microphone 113, touch screen 112, display controller 156, optical sensor 164, optical sensor controller 158, contact module 130, graphics module 132, text input module 134, contact list 137, and telephone module 138, videoconferencing module 139 includes executable instructions to initiate, conduct, and terminate a video conference between a user and one or more other participants in accordance with user instructions.

In conjunction with RF circuitry 108, touch screen 112, display controller 156, contact module 130, graphics module 132, and text input module 134, instant messaging module 141 includes executable instructions to enter a sequence of characters corresponding to an instant message, to modify previously entered characters, to transmit a respective instant message (for example, using a Short Message Service (SMS) or Multimedia Message Service (MMS) protocol for telephony-based instant messages or using XMPP, SIMPLE, or IMPS for Internet-based instant messages), to receive instant messages and to view received instant messages. In some embodiments, transmitted and/or received instant messages may include graphics, photos, audio files, video files and/or other attachments as are supported in a MMS and/or an Enhanced Messaging Service (EMS). As used herein, “instant messaging” refers to both telephony-based messages (e.g., messages sent using SMS or MMS) and Internet-based messages (e.g., messages sent using XMPP, SIMPLE, or IMPS).

In conjunction with RF circuitry 108, touch screen 112, display controller 156, contact module 130, graphics module 132, text input module 134, GPS module 135, map module 154, and music player module 146, workout support module 142 includes executable instructions to create workouts (e.g., with time, distance, and/or calorie burning goals); communicate with workout sensors (sports devices); receive workout sensor data; calibrate sensors used to monitor a workout; select and play music for a workout; and display, store and transmit workout data.

In conjunction with touch screen 112, display controller 156, optical sensor 164, optical sensor controller 158, contact module 130, graphics module 132, and image
management module 144, camera module 143 includes executable instructions to capture still images or video (includ- ing a video stream) and store them into memory 102, modify characteristics of a still image or video, or delete a still image or video from memory 102.

[0094] In conjunction with touch screen 112, display con- troller 156, contact module 130, graphics module 132, text input module 134, and camera module 143, image manage- ment module 144 includes executable instructions to arrange, modify (e.g., edit), or otherwise manipulate, label, delete, present (e.g., in a digital slide show or album), and store still and/or video images.

[0095] In conjunction with touch screen 112, display con- troller 156, contact module 130, graphics module 132, audio circuitry 110, and speaker 111, video player module 145 includes executable instructions to display, present or other- wise play back videos (e.g., on touch screen 112 or on an external, connected display via external port 124).

[0096] In conjunction with touch screen 112, display sys- tem controller 156, contact module 130, graphics module 132, audio circuitry 110, speaker 111, RF circuitry 108, and browser module 147, music player module 146 includes executable instructions that allow the user to download and play back recorded music and other sound files stored in one or more file formats, such as MP3 or AAC files. In some embodiments, device 100 may include the functionality of an MP3 player, such as an iPod (trademark of Apple Inc.).

[0097] In conjunction with RF circuitry 108, touch screen 112, display system controller 156, contact module 130, graphics module 132, and text input module 134, browser module 147 includes executable instructions to browse the Internet in accordance with user instructions, including searching, linking to, receiving, and displaying web pages or portions thereof, as well as attachments and other files linked to web pages.

[0098] In conjunction with RF circuitry 108, touch screen 112, display system controller 156, contact module 130, graphics module 132, text input module 134, e-mail client module 140, and browser module 147, calendar module 148 includes executable instructions to create, display, modify, and store calendars and data associated with calendars (e.g., calendar entries, to-do lists, etc.) in accordance with user instructions.

[0099] In conjunction with RF circuitry 108, touch screen 112, display system controller 156, contact module 130, graphics module 132, text input module 134, and browser module 147, widget modules 149 are mini-applications that may be downloaded and used by a user (e.g., weather widget 149-1, stocks widget 149-2, calculator widget 149-3, alarm clock widget 149-4, and dictionary widget 149-5) or created by the user (e.g., user-created widget 149-6). In some embodiments, a widget includes an HTML (Hypertext Markup Language) file, a CSS (Cascading Style Sheets) file, and a JavaScript file. In some embodiments, a widget includes an XML (Extensible Markup Language) file and a JavaScript file (e.g., Yahoo! Widgets).

[0100] In conjunction with RF circuitry 108, touch screen 112, display system controller 156, contact module 130, graphics module 132, text input module 134, and browser module 147, the widget creator module 150 may be used by a user to create widgets (e.g., turning a user-specified portion of a web page into a widget).

[0101] In conjunction with touch screen 112, display sys- tem controller 156, contact module 130, graphics module 132, and text input module 134, search module 151 includes executable instructions to search for text, music, sound, image, video, and/or other files in memory 102 that match one or more search criteria (e.g., one or more user-specified search terms) in accordance with user instructions.

[0102] In conjunction with touch screen 112, display con- troller 156, contact module 130, graphics module 132, and text input module 134, notes module 153 includes executable instructions to create and manage notes, to do lists, and the like in accordance with user instructions.

[0103] In conjunction with RF circuitry 108, touch screen 112, display system controller 156, contact module 130, graphics module 132, text input module 134, GPS module 135, and browser module 147, map module 154 may be used to receive, display, modify, and store maps and data associated with maps (e.g., driving directions; data on stores and other points of interest at or near a particular location; and other location-based data) in accordance with user instructions.

[0104] In conjunction with touch screen 112, display sys- tem controller 156, contact module 130, graphics module 132, audio circuitry 110, speaker 111, RF circuitry 108, text input module 134, e-mail client module 140, and browser module 147, online video module 155 includes instructions that allow the user to access, browse, receive (e.g., by streaming and/or download), play back (e.g., on the touch screen or on an external, connected display via external port 124), send an e-mail with a link to a particular online video, and other- wise manage online videos in one or more file formats, such as H.264. In some embodiments, instant messaging module 141, rather than e-mail client module 140, is used to send a link to a particular online video. Additional description of the online video application can be found in U.S. Provisional Patent Application No. 60/936,562, “Portable Multifunction Device, Method, and Graphical User Interface for Playing Online Videos,” filed Jun. 20, 2007, and U.S. patent application Ser. No. 11/968,067, “Portable Multifunction Device, Method, and Graphical User Interface for Playing Online Videos,” filed Dec. 31, 2007, the content of which is hereby incorporated by reference in its entirety.

[0105] Each of the above identified modules and applica- tions correspond to a set of executable instructions for per- forming one or more functions described above and the meth- ods described in this application (e.g., the computer- implemented methods and other information processing methods described herein). These modules (i.e., sets of instructions) need not be implemented as separate software programs, procedures or modules, and thus various subsets of these modules may be combined or otherwise re-arranged in various embodiments. For example, video player module 145 may be combined with music player module 146 into a single module (e.g., video and music player module 152, FIG. 1B). In some embodiments, memory 102 may store a subset of the modules and data structures identified above. Furthermore, memory 102 may store additional modules and data structures not described above.

[0106] In some embodiments, device 100 is a device where operation of a predefined set of functions on the device is performed exclusively through a touch screen and/or a touch- pad. By using a touch screen and/or a touchpad as the primary input control device for operation of device 100, the number of physical input control devices (such as push buttons, dials, and the like) on device 100 may be reduced.
[0107] The predefined set of functions that may be performed exclusively through a touch screen and/or a touchpad include navigation between user interfaces. In some embodiments, the touchpad, when touched by the user, navigates device 100 to a main, home, or root menu from any user interface that may be displayed on device 100. In such embodiments, the touchpad may be referred to as a "menu button." In some other embodiments, the menu button may be a physical push button or other physical input control device instead of a touchpad.

[0108] FIG. 1C is a block diagram illustrating exemplary components for event handling in accordance with some embodiments. In some embodiments, memory 102 (in FIGS. 1A and 1B) or 370 (FIG. 3) includes event sorter 170 (e.g., in operating system 126) and a respective application 136-1 (e.g., any of the aforementioned applications 137-151, 155, 380-390).

[0109] Event sorter 170 receives event information and determines the application 136-1 and application view 191 of application 136-1 to which to deliver the event information. Event sorter 170 includes event monitor 171 and event dispatcher module 174. In some embodiments, application 136-1 includes application internal state 192, which indicates the current application view(s) displayed on touch sensitive display 112 when the application is active or executing. In some embodiments, device/global internal state 157 is used by event sorter 170 to determine which application(s) is (are) currently active, and application internal state 192 is used by event sorter 170 to determine application views 191 to which to deliver event information.

[0110] In some embodiments, application internal state 192 includes additional information, such as one or more of: resume information to be used when application 136-1 resumes execution, user interface state information that indicates information being displayed or that is ready for display by application 136-1, a state queue for enabling the user to go back to a prior state or view of application 136-1, and a redo/undo queue of previous actions taken by the user.

[0111] Event monitor 171 receives event information from peripherals interface 118. Event information includes information about a sub-event (e.g., a user touch on touch-sensitive display 112, as part of a multi-touch gesture). Peripherals interface 118 transmits information it receives from I/O subsystem 106 or a sensor, such as proximity sensor 166, accelerometer(s) 168, and/or microphone 113 (through audio circuitry 110). Information that peripherals interface 118 receives from I/O subsystem 106 includes information from touch-sensitive display 112 or a touch-sensitive surface.

[0112] In some embodiments, event monitor 171 sends requests to the peripherals interface 118 at predetermined intervals. In response, peripherals interface 118 transmits event information. In other embodiments, peripheral interface 118 transmits event information only when there is a significant event (e.g., receiving an input above a predetermined noise threshold and/or for more than a predetermined duration).

[0113] In some embodiments, event sorter 170 also includes a hit view determination module 172 and/or an active event recognizer determination module 173.

[0114] Hit view determination module 172 provides software procedures for determining where a sub-event has taken place within one or more views, when touch sensitive display 112 displays more than one view. Views are made up of controls and other elements that a user can see on the display.

[0115] Another aspect of the user interface associated with an application is a set of views, sometimes herein called application views or user interface windows, in which information is displayed and touch-based gestures occur. The application views (of a respective application) in which a touch is detected may correspond to programmatic levels within a programmatic or view hierarchy of the application. For example, the lowest level view in which a touch is detected may be called the hit view, and the set of events that are recognized as proper inputs may be determined based, at least in part, on the hit view of the initial touch that begins a touch-based gesture.

[0116] Hit view determination module 172 receives information related to sub-events of a touch-based gesture. When an application has multiple views organized in a hierarchy, hit view determination module 172 identifies a hit view as the lowest view in the hierarchy which should handle the sub-event. In most circumstances, the hit view is the lowest level view in which an initiating sub-event occurs (i.e., the first sub-event in the sequence of sub-events that form an event or potential event). Once the hit view is identified by the hit view determination module, the hit view typically receives all sub-events related to the same touch or input source for which it was identified as the hit view.

[0117] Active event recognizer determination module 173 determines which view or views within a view hierarchy should receive a particular sequence of sub-events. In some embodiments, active event recognizer determination module 173 determines that only the hit view should receive a particular sequence of sub-events. In other embodiments, active event recognizer determination module 173 determines that all views that include the physical location of a sub-event are actively involved views, and therefore determines that all actively involved views should receive a particular sequence of sub-events. In other embodiments, even if touch sub-events were entirely confined to the area associated with one particular view, views higher in the hierarchy would still remain as actively involved views.

[0118] Event dispatcher module 174 dispatches the event information to an event recognizer (e.g., event recognizer 180). In embodiments including active event recognizer determination module 173, event dispatcher module 174 delivers the event information to an event recognizer determined by active event recognizer determination module 173. In some embodiments, event dispatcher module 174 stores in an event queue the event information, which is retrieved by a respective event receiver module 182.

[0119] In some embodiments, operating system 126 includes event sorter 170. Alternatively, application 136-1 includes event sorter 170. In yet other embodiments, event sorter 170 is a stand-alone module, or a part of another module stored in memory 102, such as contact/motion module 130.

[0120] In some embodiments, application 136-1 includes a plurality of event handlers 190 and one or more application views 191, each of which includes instructions for handling touch events that occur within a respective view of the application’s user interface. Each application view 191 of the application 136-1 includes one or more event recognizers 180. Typically, a respective application view 191 includes a plurality of event recognizers 180. In other embodiments, one or more of event recognizers 180 are part of a separate module, such as a user interface kit (not shown) or a higher level object from which application 136-1 inherits methods and
other properties. In some embodiments, a respective event handler 190 includes one or more of: data updater 176, object updater 177, GUI updater 178, and/or event data 179 received from event sorter 170. Event handler 190 may utilize or call data updater 176, object updater 177 or GUI updater 178 to update the application internal state 192. Alternatively, one or more of the application views 191 includes one or more respective event handlers 190. Also, in some embodiments, one or more of data updater 176, object updater 177, and GUI updater 178 are included in a respective application view 191.

A respective event recognizer 180 receives event information (e.g., event data 179) from event sorter 170, and identifies an event from the event information. Event recognizer 180 includes event receiver 182 and event comparator 184. In some embodiments, event recognizer 180 also includes at least a subset of: metadata 183, and event delivery instructions 188 (which may include sub-event delivery instructions).

Event receiver 182 receives event information from event sorter 170. The event information includes information about a sub-event, for example, a touch or a touch movement. Depending on the sub-event, the event information also includes additional information, such as location of the sub-event. When the sub-event concerns motion of a touch the event information may also include speed and direction of the sub-event. In some embodiments, events include rotation of the device from one orientation to another (e.g., from a portrait orientation to a landscape orientation, or vice versa), and the event information includes corresponding information about the current orientation (also called device attitude) of the device.

Event comparator 184 compares the event information to predefined event or sub-event definitions and, based on the comparison, determines an event or sub-event, or determines or updates the state of an event or sub-event. In some embodiments, event comparator 184 includes event definitions 186. Event definitions 186 contain definitions of events (e.g., predefined sequences of sub-events), for example, event 1 \((187-1)\), event 2 \((187-2)\), and others. In some embodiments, sub-events in an event 187 include, for example, touch begin, touch end, touch movement, touch cancellation, and multiple touching. In one example, the definition for event 1 \((187-1)\) is a double tap on a displayed object. The double tap, for example, comprises a first touch (touch begin) on the displayed object for a predetermined phase, a first lift-off (touch end) for a predetermined phase, a second touch (touch begin) on the displayed object for a predetermined phase, and a second lift-off (touch end) for a predetermined phase. In another example, the definition for event 2 \((187-2)\) is a dragging on a displayed object. The dragging, for example, comprises a touch (or contact) on the displayed object for a predetermined phase, a movement of the touch across touch sensitive display 112, and lift-off of the touch (touch end). In some embodiments, the event also includes information for one or more associated event handlers 190.

In some embodiments, event definition 187 includes a definition of an event for a respective user-interface object. In some embodiments, event comparator 184 performs a hit test to determine which user-interface object is associated with a sub-event. For example, in an application view in which three user-interface objects are displayed on touch-sensitive display 112, when a touch is detected on touch-sensitive display 112, event comparator 184 performs a hit test to determine which of the three user-interface objects is associated with the touch (sub-event). If each displayed object is associated with a respective event handler 190, the event comparator uses the result of the hit test to determine which event handler 190 should be activated. For example, event comparator 184 selects an event handler associated with the sub-event and the object triggering the hit test.

In some embodiments, the definition for a respective event 187 also includes delayed actions that delay delivery of the event information until after it has been determined whether the sequence of sub-events does or does not correspond to the event recognizer's event type.

When a respective event recognizer 180 determines that the series of sub-events do not match any of the events in event definitions 186, the respective event recognizer 180 enters an event impossible, event failed, or event ended state, after which it disregards subsequent sub-events of the touch-based gesture. In this situation, other event recognizers, if any, that remain active for the hit view continue to track and process sub-events of an ongoing touch-based gesture.

In some embodiments, a respective event recognizer 180 includes metadata 183 with configurable properties, flags, and/or lists that indicate how the event delivery system should perform sub-event delivery to actively involved event recognizers. In some embodiments, metadata 183 includes configurable properties, flags, and/or lists that indicate how event recognizers may interact with one another. In some embodiments, metadata 183 includes configurable properties, flags, and/or lists that indicate whether sub-events are delivered to varying levels in the view or programmatic hierarchy.

In some embodiments, a respective event recognizer 180 activates event handler 190 associated with an event when one or more particular sub-events of an event are recognized. In some embodiments, a respective event recognizer 180 delivers event information associated with the event to event handler 190. Activating an event handler 190 is distinct from sending (and deferred sending) sub-events to a respective hit view. In some embodiments, event recognizer 180 throws a flag associated with the recognized event, and event handler 190 associated with the flag catches the flag and performs a predefined process.

In some embodiments, event delivery instructions 188 include sub-event delivery instructions that deliver event information about a sub-event without activating an event handler. Instead, the sub-event delivery instructions deliver event information to event handlers associated with the series of sub-events or to actively involved events. Event handlers associated with the series of sub-events or with actively involved events receive the event information and perform a predetermined process.

In some embodiments, data updater 176 creates and updates data used in application 136-1. For example, data updater 176 updates the telephone number used in contacts module 137, or stores a video file used in video player module 145. In some embodiments, object updater 177 creates and updates objects used in application 136-1. For example, object updater 176 creates a new user-interface object or updates the position of a user-interface object. GUI updater 178 updates the GUI. For example, GUI updater 178 prepares display information and sends it to graphics module 132 for display on a touch-sensitive display.

In some embodiments, event handler(s) 190 includes or has access to data updater 176, object updater 177, and GUI updater 178. In some embodiments, data updater
object updater 177, and GUI updater 178 are included in a single module of a respective application 136-1 or application view 191. In other embodiments, they are included in two or more software modules.

[0132] It shall be understood that the foregoing discussion regarding event handling of user touches on touch-sensitive displays also applies to other forms of user inputs to operate multifunction devices 100 with input-devices, not all of which are initiated on touch screens, e.g., coordinating mouse movement and mouse button presses with or without single or multiple keyboard presses or holds, user movements taps, drags, scrolls, etc., on touch-pads, pen stylus inputs, movement of the device, oral instructions, detected eye movements, biometric inputs, and/or any combination thereof, which may be utilized as inputs corresponding to sub-events which define an event to be recognized.

[0133] FIG. 2 illustrates a portable multifunction device 100 having a touch screen 112 in accordance with some embodiments. The touch screen may display one or more graphics within user interface (UI) 200. In this embodiment, as well as others described below, a user may select one or more of the graphics by making contact or touching the graphics, for example, with one or more fingers 202 (not drawn to scale in the figure) or one or more styluses 203 (not drawn to scale in the figure). In some embodiments, selection of one or more graphics occurs when the user breaks contact with the one or more graphics. In some embodiments, the contact may include a gesture, such as one or more taps, one or more swipes (from left to right, right to left, upward and/or downward) and/or a rolling of a finger (from right to left, left to right, upward and/or downward) that has made contact with device 100. In some embodiments, inadvertent contact with a graphic may not select the graphic. For example, a swipe gesture that sweeps over an application icon may not select the corresponding application when the gesture corresponding to selection is a tap.

[0134] Device 100 may also include one or more physical buttons, such as “home” or menu button 204. As described previously, menu button 204 may be used to navigate to any application 136 in a set of applications that may be executed on device 100. Alternatively, in some embodiments, the menu button is implemented as a soft key in a GUI displayed on touch screen 112.

[0135] In one embodiment, device 100 includes touch screen 112, menu button 204, push button 206 for powering the device on/off and locking the device, volume adjustment button(s) 208, Subscriber Identity Module (SIM) card slot 210, head set jack 212, and docking/charging external port 124. Push button 206 may be used to turn the power on/off on the device by depressing the button and holding the button in the depressed state for a predefined time interval; to lock the device by depressing the button and releasing the button before the predefined time interval has elapsed; and/or to unlock the device or initiate an unlock process. In an alternative embodiment, device 100 also may accept verbal input for activation or deactivation of some functions through microphone 113.

[0136] FIG. 3 is a block diagram of an exemplary multifunction device with a display and a touch-sensitive surface in accordance with some embodiments. Device 300 need not be portable. In some embodiments, device 300 is a laptop computer, a desktop computer, a tablet computer, a multimedia player device, a navigation device, an educational device (such as a child’s learning toy), a gaming system, or a control device (e.g., a home or industrial controller). Device 300 typically includes one or more processing units (CPU’s) 310, one or more network or other communications interfaces 360, memory 370, and one or more communication buses 320 for interconnecting these components. Communication buses 320 may include circuitry (sometimes called a chipset) that interconnects and controls communications between system components. Device 300 includes input/output (I/O) interface 330 comprising display 340, which is typically a touch screen display. I/O interface 330 also may include a keyboard and/or mouse (or other pointing device) 350 and touchpad 355. Memory 370 includes high-speed random access memory, such as DRAM, SRAM, DDR RAM or other random access solid state memory devices; and may include non-volatile memory, such as one or more magnetic disk storage devices, optical disk storage devices, flash memory devices, or other non-volatile solid state storage devices. Memory 370 may optionally include one or more storage devices remotely located from CPU(s) 310. In some embodiments, memory 370 stores programs, modules, and data structures analogous to the programs, modules, and data structures stored in memory 102 of portable multifunction device 100 (FIG. 1), or a subset thereof. Furthermore, memory 370 may store additional programs, modules, and data structures not present in memory 102 of portable multifunction device 100. For example, memory 370 of device 300 may store drawing module 380, presentation module 382, word processing module 384, website creation module 386, disk authoring module 388, and/or spreadsheet module 390, while memory 102 of portable multifunction device 100 (FIG. 1) may not store these modules.

[0137] Each of the above identified elements in FIG. 3 may be stored in one or more of the previously mentioned memory devices. Each of the above identified modules corresponds to a set of instructions for performing a function described above. The above identified modules or programs (i.e., sets of instructions) need not be implemented as separate software programs, procedures or modules, and thus various subsets of these modules may be combined or otherwise re-arranged in various embodiments. In some embodiments, memory 370 may store a subset of the modules and data structures identified above. Furthermore, memory 370 may store additional modules and data structures not described above.

[0138] Attention is now directed towards embodiments of user interfaces (“UI”) that may be implemented on portable multifunction device 100.

[0139] FIGS. 4A and 4B illustrate exemplary user interfaces for a menu of applications on portable multifunction device 100 in accordance with some embodiments. Similar user interfaces may be implemented on device 300. In some embodiments, user interface 400 includes the following elements, or a subset or superset thereof:

- Signal strength indicator(s) 402 for wireless communication(s), such as cellular and Wi-Fi signals;
- Time 404;
- Bluetooth indicator 405;
- Battery status indicator 406;
- Tray 408 with icons for frequently used applications, such as:
  - Phone 138, which may include an indicator 414 of the number of missed calls or voicemail messages;
  - E-mail client 140, which may include an indicator 410 of the number of unread e-mails;
In some embodiments, user interface 400B includes the following elements, or a subset or superset thereof:

- 402, 404, 405, 406, 141, 148, 144, 143, 149-3, 149-2, 149-1, 149-4, 410, 414, 138, 140, and 147, as described above;
- 154, Map 154;
- 153, Notes 153;
- 412, Settings 412, which provides access to settings for device 100 and its various applications 136, as described further below;
- 155, Video and music player module 152, also referred to as iPod (trademark of Apple Inc.) module 152; and
- 155, Online video module 155, also referred to as YouTube (trademark of Google Inc.) module 155.

FIG. 4C illustrates an exemplary user interface on a device (e.g., device 300, FIG. 3) with a touch-sensitive surface 451 (e.g., a tablet or touchpad 355, FIG. 3) that is separate from the display 450 (e.g., touch screen display 112). Although many of the examples which follow will be given with reference to inputs on touch screen display 112 (where the touch sensitive surface and the display are combined), in some embodiments, the device detects inputs on a touch-sensitive surface that is separate from the display, as shown in FIG. 4C. In some embodiments the touch sensitive surface (e.g., 451 in FIG. 4C) has a primary axis (e.g., 452 in FIG. 4C) that corresponds to a primary axis (e.g., 453 in FIG. 4C) on the display (e.g., 450). In accordance with these embodiments, the device detects contacts (e.g., 460 and 462 in FIG. 4C) with the touch-sensitive surface 451 at locations that correspond to respective locations on the display (e.g., in FIG. 4C 460 corresponds to 468 and 462 corresponds to 470). In this way, user inputs (e.g., contacts 460 and 462, and movements thereof) detected by the device on the touch-sensitive surface (e.g., 451 in FIG. 4C) are used by the device to manipulate the user interface on the display (e.g., 450 in FIG. 4C) of the multifunction device when the touch-sensitive surface is separate from the display. It should be understood that similar methods may be used for other user interfaces described herein.

Attention is now directed towards embodiments of user interfaces ("UI") and associated processes that may be implemented on a multifunction device with a display and a touch-sensitive surface, such as device 300 or portable multifunction device 100.

FIGS. 5A-5J illustrate exemplary user interfaces for displaying menus in accordance with some embodiments. The user interfaces in these figures are used to illustrate the processes described below, including the processes in FIGS. 6A-6C.

In FIGS. 5A-5J, the size of finger contacts or the distance of movements may be exaggerated for illustrative purposes. No depiction in the figures bearing on sizes or movements of finger contacts should be taken as a requirement or limitation for the purpose of understanding sizes and scale associated with the methods and devices disclosed herein.

FIGS. 5A-5E illustrate exemplary user interfaces for displaying a respective menu at a fixed angle (or orientation), in accordance with some embodiments. FIG. 5A depicts an exemplary user interface displaying user interface objects 502, 504, and 506 on touch screen 112 of multifunction device 100. FIG. 5A also illustrates detection of finger contact 505 on touch screen 112 at location 505-A away from any of the displayed user interface objects (e.g., contact 505 does not overlap with any of the displayed user interface objects 502, 504, and 506).

As illustrated, finger contact 505 typically has a shape of an ellipse (or an oval). An ellipse (e.g., area of contact 505) has a major axis (e.g., 512-1) and a minor axis (e.g., 512-2). In some cases, an angle between a predefined reference axis (e.g., vertical display axis 514) and the major axis (e.g., 512-1) or minor axis (e.g., 512-2) is used to characterize the elliptical contact area. For example, 516, which represents an angle between major axis 512-1 and vertical display axis 514, is used to characterize the elliptical contact area (e.g., when 516 ranges from −90 degrees to 90 degrees, 516 is a positive number when contact 505 tilts clockwise from a vertical line as shown in FIG. 5A, and a negative number when contact 505 tilts counterclockwise from the vertical line (not shown)). In some embodiments, instead of the major axis or the minor axis, another axis (e.g., 512-3) is used to characterize the elliptical contact area. Axis 512-3 is neither a major axis nor a minor axis (e.g., axis 512-3 is an axis angled 45 degrees from the major axis).

FIG. 5A also illustrates that menu 508-1 is displayed in response to the detection of contact 505 on touch screen 112. In this example, menu 508-1 includes a plurality of menu icons 510-1 through 510-3. A menu icon 510, when activated (e.g., by a finger gesture, such as a tap gesture), initiates a corresponding operation. In some embodiments, the menu comprises a color palette menu, and each menu icon in the color palette menu corresponds to a respective color. A menu icon in the color palette menu, when activated (e.g., by a finger gesture), changes a color of one of: foreground (e.g., a color to be used for drawing), background, user interface object, pattern (e.g., a fill) within a user interface object, and/or shadow. In some embodiments, the menu comprises a tool palette menu, and each menu icon in the tool palette menu comprises a different drawing tool (e.g., line, rectangle, circle/ellipse, polyline, text, block, smudge, color picker, etc.). For example, a line tool when activated enables drawing of a line in a drawing application. A rectangle tool when activated enables drawing of a rectangle in a drawing application. In some embodiments, the menu comprises a brush menu, and the brush menu includes a plurality of menu icons that correspond to various widths and/or shapes of brushes. In some embodiments, the menu comprises a file menu, and each menu icon in the file menu corresponds to one of: new file, file open, and file save functions.

In FIG. 5B, rotation of contact 505 is detected on touch screen 112 at location 505-D. While contact 505 rotates, contact 505 remains in contact with touch screen 112. In some embodiments, the rotation of contact 505 can be
detected by using an angle of major axis 512-1, minor axis 512-2, and/or oblique axis (e.g., 512-3 shown in FIG. 5A) relative to a predefined reference axis (e.g., vertical display axis 514 shown in FIG. 5A). In some embodiments, an absolute angle of major axis 512-2, minor axis 512-2, and/or an oblique axis is used to detect the rotation of contact 505. In some embodiments, a change in the angle (e.g., compared to an angle detected before) of major axis 512-2, minor axis 512-2, and/or an oblique axis is used to detect the rotation of contact 505. In FIGS. 5A-5B, contact 505 rotated counterclockwise. In response to the detection of rotation of contact 505, menu 508-1 ceases to be displayed and menu 508-2 is displayed. In this example, menu 508-2 includes a plurality of menu icons 510-4 through 510-6.

[0177] FIG. 5C illustrates additional rotation of contact 505 on touch screen 112 at location 505-C. While contact 505 rotates, contact 505 remains in contact with touch screen 112. In FIGS. 5A-5C, contact 505 rotated counterclockwise. In response to detecting the rotation of contact 505, menu 508-2 ceases to be displayed and menu 508-3 is displayed. In this example, menu 508-3 includes a plurality of menu icons 510-7 and 510-8.

[0178] In FIG. 5D, contact 505 moves across touch screen 112 to location 505-D, away from any of the displayed user interface objects. While contact 505 moves across touch screen 112, contact 505 remains in contact with touch screen 112. As illustrated, menu 508-3 moves in accordance with the movement of contact 505.

[0179] FIG. 5E illustrates contact 505 moving across touch screen 112 to location 505-E, which corresponds to a location of user interface object 506. While contact 505 moves across touch screen 112, contact 505 remains in contact with touch screen 112. As illustrated, menu 508-3 continues to be displayed on touch screen 112, and moves in accordance with the movement of contact 505. However, in some embodiments, menu 508-3 ceases to be displayed when contact 505 moves across touch screen 112 to a location that corresponds to a location of a displayed user interface object.

[0180] FIGS. 5F-5G illustrate exemplary user interfaces for displaying a respective menu at a respective angle, in accordance with some embodiments.

[0181] FIG. 5F illustrates detection of contact 507 on touch screen 112 at location 507-A, away from any of the displayed user interface objects. FIG. 5F also illustrates that menu 508-1 is displayed in response to the detection of contact 507 on touch screen 112.

[0182] In FIG. 5G, rotation of contact 507 is detected on touch screen 112 at location 507-B, away from any of the displayed user interface objects. In response to the detection of rotation of contact 507, menu 508-1 ceases to be displayed and menu 508-2 is displayed. As illustrated, menu 508-2 is displayed at an angle (or orientation) that is distinct from an angle (or orientation) of menu 508-1 (shown in FIG. 5F).

[0183] In some embodiments, the angle of menu 508-2 is determined based on an absolute angle of contact 507 or a relative angle of contact 507. In some embodiments, the orientation of a respective menu corresponds to an orientation of the finger contact. In other embodiments, the orientation of a respective menu corresponds to a change in the angle of contact 507 (e.g., when contact 507 rotates 20 degrees, a second menu (e.g., 508-2) is displayed at, for example, a 10 degree angle, which is half of 20 degrees). In some embodiments, a respective menu can be rotated continuously (or pseudo-continuously) within a range in accordance with rotation of the finger contact. In such embodiments, menu 508-1 is displayed at various angles (e.g., in one degree increments) within a first range (e.g., between zero degrees and twenty degrees) in accordance with rotation of contact 507, and menu 508-2 is displayed at various angles within a second range (e.g., between twenty degrees and forty degrees) in accordance with rotation of contact 507. In other embodiments, a respective menu is displayed at a respective distinct angle. In such embodiments, menu 508-1 is displayed at a first predefined angle (e.g., zero degree), and menu 508-2 is displayed at a second predefined angle that is distinct from the first predefined angle (e.g., twenty degrees).

[0184] FIGS. 5I-5J illustrate exemplary user interfaces for displaying a respective menu, in accordance with some embodiments.

[0185] FIG. 5I illustrates detection of contact 509 on touch screen 112 at location 509-A, away from any of the displayed user interface objects. FIG. 5I also illustrates that menu 508-4 is displayed in response to the detection of contact 509 on touch screen 112. Menu 508-4 is displayed at a location that is proximate to the location of contact 509 (e.g., within 100 pixels, 50 pixels, or 20 pixels; or within one inch, a half inch, or a quarter inch, but menu 508-4 does not overlap with the location of contact 509.

[0186] In FIG. 5J, rotation of contact 509 is detected on touch screen 112 at location 509-B. In response to detecting of rotation of contact 509, menu 508-4 ceases to be displayed and menu 508-5 is displayed. As illustrated, menu 508-5 is displayed at a radial location that is distinct from a position of menu 508-4 (shown in FIG. 5I).

[0187] FIG. 5J illustrates detection of contact 511 on touch screen 112 at a location that is away from any of the displayed user interface objects. FIG. 5J also illustrates that menu 508-6 is displayed in response to the detection of contact 511 on touch screen 112. As illustrated, menu 508-6 includes menu icons 510-13 and 510-14. Menu 508-6 is displayed at a central location of touch screen 112. For example, menu 508-6 is displayed at or near (e.g., within 100 pixels, 50 pixels, or 20 pixels; within one inch, a half inch, or a quarter inch; or within one tenth of a width or height of touch screen 112) a center of touch screen 112. As contact 511 rotates, menu 508-6 is replaced by other menus in the center of touch screen 112 (not shown).

[0188] FIGS. 6A-6C are flow diagrams illustrating method 600 of displaying menus in accordance with some embodiments. Method 600 is performed at a multifunction device (e.g., device 300, FIG. 3, or portable multifunction device 100, FIG. 1) with a display and a touch-sensitive surface. In some embodiments, the display is a touch screen display and the touch-sensitive surface is on the display. In some embodiments, the display is separate from the touch-sensitive surface. Some operations in method 600 may be combined and/or the order of some operations may be changed.

[0189] As described below, method 600 provides an intuitive way to display a plurality of menus in response to detecting rotation of a finger contact. The method reduces the cognitive burden on a user when displaying menus, thereby creating a more efficient human-machine interface. For battery-operated computing devices, enabling a user to display
menus faster and more efficiently conserves power and increases the time between battery charges.

[0190] The device displays (602) one or more user interface objects on the touch-sensitive display (e.g., user interface objects 502, 504, and 506 in FIG. 5A).

[0191] The device detects (604) a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects (e.g., contact 505 in FIG. 5A) is away from any of displayed user interface objects 502, 504, and 506). In some embodiments, a finger contact is away from any of the displayed user interface objects when a location of the finger contact does not correspond to a respective location of any of the displayed user interface objects. In some embodiments, a finger contact is away from any of the displayed user interface objects when a location of the finger contact does not correspond to a hidden hit region (a non-displayed region which when selected by a finger contact activates a corresponding user interface object) of any of the displayed user interface objects. In some embodiments, a finger contact is away from any of the displayed user interface objects when detection of the finger contact does not activate any function (e.g., selecting, moving, etc.) associated with any of the displayed user interface objects. As noted above, this allows display of a menu in response to detecting a contact that does not correspond to a particular user interface object, thereby eliminating the need for touching the location of a menu object.

[0192] The device determines (606) whether a predefined condition is satisfied. The predefined condition includes that the finger contact (e.g., 505 in FIG. 5A) continues to be detected on the touch-sensitive display (e.g., 112).

[0193] In some embodiments, the predefined condition includes (608) that the finger contact is made with a finger of a non-dominant hand. As used herein, a finger includes any digit of a hand, including a thumb, an index finger, a middle finger, a ring finger, and a little finger.

[0194] In some embodiments, a non-dominant hand (or a non-dominant hand) is determined based on a prior user input (e.g., a user indicates whether a left or right hand is a dominant hand or a non-dominant hand in a preference setting). In some embodiments, a non-dominant hand (or a dominant hand) is determined based on a frequency of use of a left hand finger or a right hand finger detected by the device. In some embodiments, a non-dominant hand (or a dominant hand) is determined by default in the absence of a prior user input.

[0195] In some embodiments, whether a finger contact is made with a finger of a non-dominant hand is determined based on an angle of the contact. In some embodiments, when a finger contact is detected, an ellipse is fitted to an area of the finger contact, and an angle of the ellipse (e.g., an angle of the major axis, minor axis, or any other axis of the ellipse) is used to determine whether the finger contact is made with a finger of a non-dominant hand. For example, as illustrated in FIG. 5A, angle $\theta_{516}$ between major axis 512-1 and vertical axis 514 can represent whether the finger contact is made with a finger of a non-dominant hand. When angle $\theta_{516}$ is positive (e.g., the ellipse is angled toward right as shown in FIG. 5A), the finger contact is deemed to be made with a finger of a left hand. When angle $\theta_{516}$ is negative (e.g., the ellipse is angled toward left as shown in FIG. 5C), the finger contact is deemed to be made with a finger of a right hand. In addition, the identity of a non-dominant hand (e.g., indicated by a prior user input) is used to determine whether the finger contact is made with a finger of a non-dominant hand. In some embodiments, the initial angle of the ellipse (determined when the finger contact is first detected) is used to determine whether the finger contact is made with a finger of a non-dominant hand, independent of any subsequent changes to the angle of the ellipse (or subsequent rotation of the finger contact) such that any subsequent rotation of the finger contact does not alter the identity of a hand that is associated with the finger contact. In some embodiments, any combination of methods described above can be used.

[0196] In some embodiments, the predefined condition includes (610) that the finger contact is made with a thumb of a non-dominant hand. In some embodiments, the predefined condition includes that the finger contact is made with a non-thumb finger of a non-dominant hand. In some embodiments, whether the finger contact is made with a thumb or a non-thumb finger is determined based on an ellipticity of the ellipse (e.g., a ratio of a major axis length and a minor axis length) that is fitted to an area of the finger contact. In some embodiments, whether the finger contact is made with a thumb or a non-thumb finger is determined based on an area of the finger contact (e.g., a finger contact made with a thumb has a larger area than a finger contact made with a non-thumb finger). In some embodiments, any combination of methods described above can be used.

[0197] Operations 614-640 are performed while the predefined condition is satisfied (612).

[0198] The device displays (614) a first menu of a plurality of menus (e.g., at a location of the finger contact, as shown in FIG. 5A: proximate to the location of the finger contact, as shown in FIG. 5B; or at a central location of the touch-sensitive display, as shown in FIG. 5I). In some embodiments, the first menu is displayed at a predetermined location (e.g., near the top of touch screen 112 or along one side of touch screen 112).

[0199] The device detects (616) a rotation of the finger contact. In some embodiments, detecting the rotation of the finger contact includes detecting an angle of the finger contact. In some embodiments, the angle of the finger contact is determined with reference to a predefined axis of the display. For example, as shown in FIG. 5A, the angle of finger contact 505 is determined based on a major axis of an ellipse that is fitted to finger contact 505, with reference to a predefined axis (e.g., vertical axis 514) of the display. Alternatively, a horizontal axis or any other axis of the display can be used as a reference. In some embodiments, the rotation of the finger contact is detected when the angle of the finger contact changes with reference to the predefined axis of the display. In some embodiments, the rotation of the finger contact is detected when the angle of the finger contact differs by more than a predefined threshold (e.g., one degree; five degrees; or ten degrees) with reference to a previously detected angle of the finger contact.

[0200] In some embodiments, the device determines (620) a touch area that corresponds to the finger contact on the touch-sensitive display. In some embodiments, the device fits an ellipse (or an oval) to an area of the finger contact detected on the touch-sensitive display.

[0201] In some embodiments, the device determines (622) a major axis of the touch area (e.g., 512-1 in FIG. 5A). Detecting the rotation of the finger contact includes detecting a rotation of the major axis of the touch area.

[0202] In some embodiments, the device determines (630) a minor axis of the touch area (e.g., 512-2 in FIG. 5A).
Detecting the rotation of the finger contact includes detecting a rotation of the minor axis of the touch area.

[0203] In some embodiments, the device determines a predefined axis of the touch area that is neither a major axis nor a minor axis, such as axis 512-3 in FIG. 5A. Detecting the rotation of the finger contact includes detecting a rotation of the predefined axis of the touch area.

[0204] The device replaces (618) display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact. In some embodiments, the device displays the second menu of the plurality of menus in accordance with a current angle of the finger contact with respect to a predefined axis of the display. For example, the device displays: menu 508-1 when the angle of the major axis of finger contact 505 is within 10 to 30 degrees of vertical axis 514; menu 508-2 when the angle of the major axis of finger contact 505 is within -10 to 10 degrees of vertical axis 514; and menu 508-3 when the angle of the major axis of finger contact 505 is within -30 to -10 degrees of vertical axis 514. In some embodiments, the device displays the second menu of the plurality of menus in accordance with a change in the angle of the finger contact. In some embodiments, the device displays a series of menus in the plurality of menus in accordance with the rotation of the finger contact (e.g., a series of menus are displayed in sequence as the finger contact rotates). Menus in the series of menus may or may not have a same number of menu icons. Menus in the series of menus may or may not be of a same size. As noted above, this method streamlines the menu selection process by displaying the second menu in response to a rotation of the contact, thereby eliminating the need to translate the contact or use multiple contacts to navigate through multiple menus.

[0205] In some embodiments, displaying the second menu includes (632) displaying the second menu at a respective angle in accordance with the rotation of the finger contact (e.g., menu 508-2 in FIG. 5G).

[0206] In some embodiments, displaying the second menu includes (634) displaying the second menu at a respective radial location relative to a location of the finger contact in accordance with the rotation of the finger contact (e.g., menu 508-4 in FIG. 5I and menu 508-5 in FIG. 5I). In some embodiments, a respective radial location is located at a predefined location of the finger contact. In some embodiments, two adjacent radial locations are separated by a predefined angle (e.g., 15, 20, 30, or 45 degrees).

[0207] In some embodiments, displaying the second menu of the plurality of menus includes (636) displaying the second menu of the plurality of menus independent of a location of the finger contact on the touch-sensitive display (e.g., in FIGS. 5C-SE, menu 508-3 is displayed independent of a location of finger contact 505). In some embodiments, the device displays the second menu of the plurality of menus even if the location of the finger contact, after displaying the first menu of the plurality of menus, corresponds to one of the displayed user interface objects.

[0208] In some embodiments, displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact includes (638) displaying the second menu of the plurality of menus at a location of the finger contact (e.g., in FIG. 5I, menu 508-2 overlaps with location 505-B of finger contact 505).

[0209] In some embodiments, displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact includes (640) displaying the second menu of the plurality of menus proximate to the location of the finger contact (e.g., in FIG. 5I, menu 508-5 is displayed proximate to the location of finger contact 505).

[0210] In some embodiments, displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact includes (642) displaying the second menu of the plurality of menus at a central location of the touch-sensitive display (e.g., menu 508-6 in FIG. 5I).

[0211] In some embodiments, the device detects (620) a movement of the finger contact across the touch-sensitive display, and moves the second menu in accordance with the movement of the finger contact (e.g., in FIGS. 5C-SE, menu 508-3 is moved in accordance with the movement of finger contact 505).

[0212] In some embodiments, while the predefined condition is satisfied, the device detects (622) a second rotation of the finger contact, and displays a third menu of the plurality of menus in accordance with the second rotation of the finger contact (e.g., in FIG. 5C, menu 508-3 is displayed in response to further rotation of finger contact 505). The second rotation of the finger contact may be in the same direction as the previous rotation of the finger contact (e.g., a counterclockwise rotation of contact 505 in FIGS. 5I-5C compared to a counterclockwise rotation of contact 505 in FIGS. 5A-5I), or in the opposite direction as the previous rotation of the finger contact (e.g., a clockwise rotation of contact 505, not shown). In some embodiments, while the predefined condition is satisfied, the device continues to detect rotation of the finger contact, and displays a respective menu of the plurality of menus in accordance with the rotation of the finger contact.

[0213] In some embodiments, the device detects (624) activation of an operation in the second menu (e.g., via selection of a menu icon 510 in the second menu with a finger tap gesture). In response, the device performs a corresponding operation (not shown).

[0214] The operations in the information processing methods described above may be implemented by running one or more functional modules in information processing apparatus such as general purpose processors or application specific chips. These modules, combinations of these modules, and/or their combination with general hardware (e.g., as described above with respect to FIGS. 1A, 1B and 3) are all included within the scope of protection of the invention.

[0215] The operations described above with reference to FIGS. 6A-6C may be implemented by components depicted in FIGS. 1A-1C. For example, detection operation 604, display operation 618, and move operation 620 may be implemented by event sorter 170, event recognizer 180, and event handler 190. Event monitor 171 in event sorter 170 detects a contact on touch-sensitive display 112, and event dispatcher module 174 delivers the event information to application 136-1. A respective event recognizer 180 of application 136-1 compares the event information to respective event definitions 186, and determines whether a first contact at a first location on the touch-sensitive surface corresponds to a predefined event or sub-event, such as selection of an object on a user interface. When a respective predefined event or sub-event is detected, event recognizer 180 activates an event handler 190 associated with the detection of the event or sub-event. Event handler 190 may utilize or call data updater 176 or object updater 177 to update the application internal state 192. In some embodiments, event handler 190 accesses a respective GUI updater 178 to update what is displayed by
the application. Similarly, it would be clear to a person having ordinary skill in the art how other processes can be implemented based on the components depicted in FIGS. 1A-1C.

The foregoing description, for purpose of explanation, has been described with reference to specific embodiments. However, the illustrative discussions above are not intended to be exhaustive or to limit the invention to the precise forms disclosed. Many modifications and variations are possible in view of the above teachings. The embodiments were chosen and described in order to best explain the principles of the invention and its practical applications, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated.

What is claimed is:

1. An electronic device, comprising:
   a touch-sensitive display;
   one or more processors;
   memory; and
   one or more programs, wherein the one or more programs are stored in the memory and configured to be executed by the one or more processors, the one or more programs including instructions for:
   displaying one or more user interface objects on the touch-sensitive display;
   detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects;
   determining whether a predefined condition is satisfied, the predefined condition including that the finger contact continues to be detected on the touch-sensitive display; and
   while the predefined condition is satisfied:
   displaying a first menu of a plurality of menus;
   detecting a rotation of the finger contact; and
   replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

2. The device of claim 1, wherein the predefined condition includes that the finger contact is made with a finger of a non-dominant hand.

3. The device of claim 1, wherein the predefined condition includes that the finger contact is made with a thumb of a non-dominant hand.

4. The device of claim 1, wherein the one or more programs include instructions for determining a touch area that corresponds to the finger contact on the touch-sensitive display.

5. The device of claim 4, wherein the one or more programs include instructions for determining a major axis of the touch area, and the instructions for detecting the rotation of the finger contact include instructions for detecting a rotation of the major axis of the touch area.

6. The device of claim 4, wherein the one or more programs include instructions for determining a minor axis of the touch area, and the instructions for detecting the rotation of the finger contact include instructions for detecting a rotation of the minor axis of the touch area.

7. The device of claim 1, wherein the one or more programs include instructions for:
   detecting a movement of the finger contact across the touch-sensitive display; and
   moving the second menu in accordance with the movement of the finger contact.

8. The device of claim 1, wherein instructions for displaying the second menu include instructions for displaying the second menu at a respective angle in accordance with the rotation of the finger contact.

9. The device of claim 1, wherein instructions for displaying the second menu include instructions for displaying the second menu at a respective radial location relative to a location of the finger contact in accordance with the rotation of the finger contact.

10. The device of claim 1, wherein instructions for displaying the second menu of the plurality of menus include instructions for displaying the second menu of the plurality of menus independent of a location of the finger contact on the touch-sensitive display.

11. The device of claim 1, wherein instructions for displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact include instructions for displaying the second menu of the plurality of menus at a location of the finger contact.

12. The device of claim 1, wherein instructions for displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact include instructions for displaying the second menu of the plurality of menus proximate to the location of the finger contact.

13. The device of claim 1, wherein instructions for displaying the second menu of the plurality of menus in accordance with the rotation of the finger contact include instructions for displaying the second menu of the plurality of menus at a central location of the touch-sensitive display.

14. The device of claim 1, wherein the one or more programs include instructions for:
   while the predefined condition is satisfied:
   detecting a second rotation of the finger contact; and
   replacing display of the second menu with display of a third menu of the plurality of menus in accordance with the second rotation of the finger contact.

15. A method, comprising:
   at an electronic device with a touch-sensitive display:
   displaying one or more user interface objects on the touch-sensitive display;
   detecting a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects;
   determining whether a predefined condition is satisfied, the predefined condition including that the finger contact continues to be detected on the touch-sensitive display; and
   while the predefined condition is satisfied:
   displaying a first menu of a plurality of menus;
   detecting a rotation of the finger contact; and
   replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

16. The method of claim 15, wherein the predefined condition includes that the finger contact is made with a finger of a non-dominant hand.

17. The method of claim 15, including:
   detecting a movement of the finger contact across the touch-sensitive display; and
   moving the second menu in accordance with the movement of the finger contact.

18. The method of claim 15, wherein displaying the second menu includes displaying the second menu at a respective angle in accordance with the rotation of the finger contact.
19. The method of claim 15, wherein displaying the second menu includes displaying the second menu at a respective radial location relative to a location of the finger contact in accordance with the rotation of the finger contact.

20. A computer readable storage medium storing one or more programs, the one or more programs comprising instructions, which when executed by an electronic device with a touch-sensitive display, cause the device to:
   display one or more user interface objects on the touch-sensitive display;
   detect a finger contact on the touch-sensitive display at a first location away from any of the displayed user interface objects;
   determine whether a predefined condition is satisfied, the predefined condition including that the finger contact continues to be detected on the touch-sensitive display; and
   while the predefined condition is satisfied:
   display a first menu of a plurality of menus;
   detect a rotation of the finger contact; and
   replacing display of the first menu with display of a second menu of the plurality of menus in accordance with the rotation of the finger contact.

* * * * *